




EX LIBRIS
UNIVERSITATIS
ALBERTÆNSIS



Digitized by the Internet Archive
in 2025 with funding from
University of Alberta Library

<https://archive.org/details/0162009261692>

UNIVERSITY OF ALBERTA

LIBRARY RELEASE FORM

NAME OF AUTHOR: Hong Zhu

TITLE OF THESIS: Dynamic Programming Algorithm for Segmentation of CVC Syllables

DEGREE: Master of Science

YEAR THIS DEGREE GRANTED: 1998

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly, or scientific purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as hereinbefore provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

UNIVERSITY OF ALBERTA

Dynamic Programming Algorithm for Segmentation of CVC Syllables

by



Hong Zhu

A thesis submitted to the Faculty of Graduate Studies and Research in
partial fulfillment of the requirements for the degree of Master of Science

in

Speech Production and Perception

Department of Linguistics

Edmonton, Alberta

Spring 1998

THE UNIVERSITY OF ALBERTA
FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled DYNAMIC PROGRAMMING ALGORITHMS FOR SEGMENTATION OF CVC SYLLABLES submitted by Hong Zhu in partial fulfillment of the requirements for the degree of Master of Science in Speech Production and Perception.

For My Grandmother

ABSTRACT

This research is a study of an autosegmentation procedure for parsing English CVC syllables into subphoneme-sized units -- microsegments. The implementation of the procedure makes use of the techniques in HMM and dynamic programming methods. An HMM-like model is set up with the microsegments being the states of the model. The whole procedure consists of two stages. First, a training stage is conducted for the model parameters. Then, a testing stage involves a Viterbi search to put the boundaries for the microsegments. To improve the model performance, a series of experiments are conducted under different conditions to find out the 'best' model architecture.

ACKNOWLEDGMENTS

I gratefully thank my supervisor Dr. Terrance Nearey for his guidance and help during the period of my research work for my master's thesis. I would also like to thank my committee members Dr. Bernard Rochet for his interests in and contributions to my work and Dr. Megan Hodge for her advice and suggestions.

I thank the department of linguistics, University of Alberta for assisting me to finish my Master's degree and I thank all the professors who have educated me through all these years.

I would also like to show my appreciation to my colleagues Dr. Michael Kiefte for discussion of his previous work in this area and Dr. Fangxing Chen for his help and kindness.

Finally, I would like to show my respects to Dr. Bruce Derwing.

TABLE OF CONTENTS

Chapter	Page
I. Introduction	1
1. A Review of Automatic Speech Segmentation	1
Why is Segmentation Necessary?	1
Basic Pattern Recognition Techniques in Speech	2
Dynamic Time Warping	4
Hidden Markov Models	4
Artificial Neural Networks	5
Survey of Existing Segmentation Techniques	6
Two Categories in Speech Segmentation	6
Unconstrained Acoustic Segmentation	7
Constrained Segmentation	9
Segmentation Techniques	12
Model-free Method	12
Model-tied Method	13
2. Continuous Variable Duration Hidden (Semi-)	
Markov Model	14

HMM Used for Modeling Speech	14
Discrete HMM	16
Continuous HMM	17
Semi-continuous HMM	19
State Durations	19
3. A Statement of the Research	24
II. Description of the Implementation Procedure	26
1. Introduction	26
Microsegments Used in This Research	26
2. The Implementation Procedure	28
Theoretical Basis for the Set Up of the Model	28
Certain Modifications Adopted in This Research	30
3. Data	32
Training and Testing data	32
4. Extracting Feature Vectors	35
“Standard” Feature Representations	35
The Calculation of Feature Vector in This Research .	36

Continuation Part of the Microsegment	39
5. Transitions of the Model States	41
6. Modeling of State Duration	44
7. Modeling of Final Segments	45
8. Observation Distributions	45
9. Viterbi Search	46
III. Segmentation Experiments and Analysis	50
1. Introduction	50
2. Two Stages of Experiments	51
Training	51
Testing	52
3. Segmentation Experiments and Results	54
Segmentation Experiment 1	54
Experiment 2 -- Segmentation of Final Consonant ...	56
Experiment 3 -- Reduced Number of	
Cepstral Coefficients	59
Experiment 4 -- Introduction of	
Delta-cepstral Coefficients	61

4. Discussion and Conclusion	64
IV. Summary	66
References	70

LIST OF TABLES

Table	Description	Page
1.	Results of Wang's experiment	21
2.	Hand-marked cursors and their descriptions	37
3.	Basic signal types defined by feature vectors	42
4.	Results of Experiment 1	54
5.	Results of segmentation without modeling the final segments	57
6.	Experiment with 9 cepstral coefficients	59
7.	Experiment with delta-cepstral coefficients	62

LIST OF FIGURES

Figure	Description	page
1.	Steps taken in pattern recognition approach	3
2.	Segmentation template matching	11
3.	Illustration of general interstate connection	22
4.	Signals with hand-marked cursors	36
4.	Locations of onset <i>vs</i> continuation part of the signal	41
6.	Possible state transitions	43
7.	Possible transition of states for final segments	44
8.	Boxplot for Experiment 1	55
9.	Boundaries for the final consonant	57
10.	Boxplot of Experiment 2	58
11.	Boxplot of Experiment 3	60
12.	Boxplot of Experiment 4	63

CHAPTER 1 INTRODUCTION

This chapter introduces some current developments in the area of segmentation of speech signals to which the present study is related. It also specifies the subject matter of the study, its theoretical basis, and the implementation of the research.

1. A Review of Automatic Speech Segmentation

1.1 Why is Segmentation Necessary?

The task of speech recognition is to take the acoustic waveform produced by the speaker as an input and output a sequence of linguistic words corresponding to the input. In the early stages of research on speech recognition, the method is to develop statistical models for each word in a recognizer vocabulary and to use a standard pattern recognition technique to recognize speech. However, this method is effective only for a relatively small vocabulary due to the large amount of training data required. Most larger vocabulary systems now make use of speech units smaller than a word so that different words can share common segments for their representations. Since such smaller units occur frequently in a training set, more reliable

estimates of statistical properties can be obtained. But such units can not generally be uttered in isolation and have to be extracted from a continuous stream of acoustic parameters. This is done by the process of segmentation.

Segmentation and labeling are also very important in speech processing for the scientific analysis of speech sounds. In speech synthesis, a dictionary of subword units also has to be built up on the basis of segmenting large units into elementary units.

One approach to segmentation is to do it manually. But this is very time consuming because it requires a large amount of work in listening and waveform interpretation. In addition, this kind of work has to be conducted by phonetically skilled people and the decisions they make may sometimes be subjective and inconsistent. Therefore, development of automatic segmentation algorithms is highly desirable.

1.2 Basic Pattern Recognition Techniques in Speech

The basic segmentation problem falls within the range of pattern recognition. The aim of pattern recognition is to use a mathematical approach to make decisions (in a probabilistic sense) about which category the observation sequence belongs to. These decisions are based on either *a posteriori* knowledge obtained from training data or prior knowledge of the

categories, or both (Figure 1). In an autosegmentation system, the acoustic events are taken as inputs of the pattern recognition system, and the outputs are the segment boundaries. Such a system includes either some prior knowledge, or contains a training that provides the system with certain *a posteriori* knowledge. Segmentation by a system using a training session to provide *a posteriori* knowledge is called supervised segmentation. The training session provides the system with category information through manual segmentation of the training data. In this case, only probabilistic structure is learnt. A system without a training phase implements unsupervised segmentation, in which no category information is available.

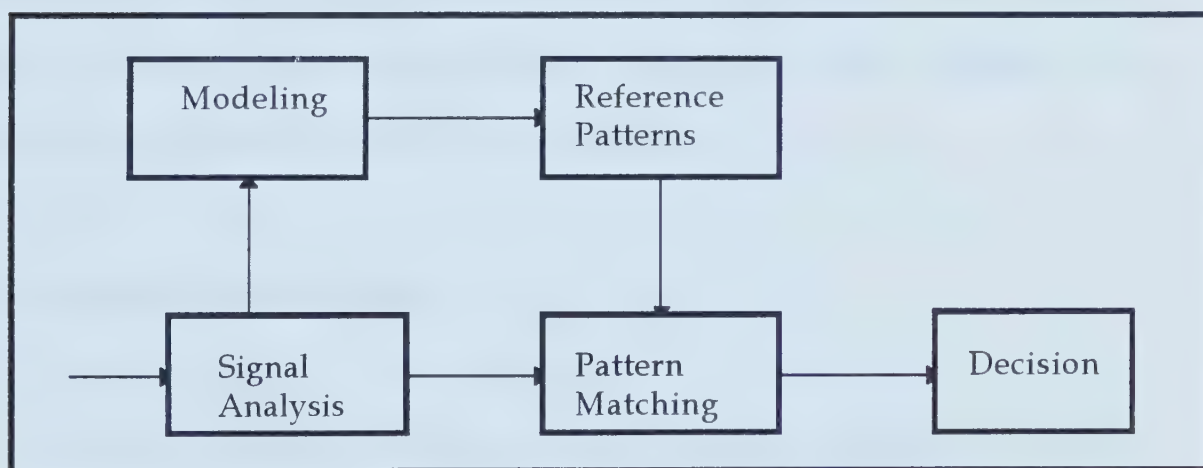


Figure 1 Steps taken in pattern recognition approach
(After Juang, Perdue & Thompson, 1995)

1 Dynamic Time Warping

In the area of speech recognition, there are three types of acoustic pattern matching techniques. The first one is called dynamic time warping (DTW). Given two acoustic patterns: $X = x_1, x_2, \dots, x_n$ and $Y = y_1, y_2, \dots, y_m$, in which n and m are total number of frames of X and Y , and x_i is the i th frame of X , which maybe a vector of bandpass filter outputs, or a set of cepstral coefficients, a path is found which minimizes the distance between the two patterns. The technique used to find the path is called dynamic programming. The key to dynamic programming is to find the optimum global path by always choosing at each point in time the path which minimizes the accumulated distance between the two patterns.

1.2.2. Hidden Markov Models

Another acoustic pattern matching technique utilizes the hidden Markov model (HMM). HMM is a technique to model speech signals as an output of a stochastic process. It consists of a Markov chain with certain number of states. At one time instant, the model would stay in a state, until next time instant, it would move to a new state or stay in the same state. The procedure is repeated until a complete sequence has been produced. There is a

probability $\{a_{ij}\}$ associated with each transition. The starting state is also uncertain and governed by initial state probabilities $\{\pi_i\}$. The model is said to be 'hidden' in that the observation sequence is one of a finite set of patterns which is also determined by a probability $\{b_{jk}\}$. What is hidden is the state sequence.

1.2.3 Artificial Neural Networks

A newly developed pattern recognition technique now gaining prominence involves neural networks. Artificial neural networks are pattern-matching devices with processing mechanisms broadly inspired by the structure of the biological nervous system. They are interesting devices which can learn general characteristics or rules from limited sample data. The network needs to be trained and an unknown input is calculated to give an output result. This output is compared with certain output patterns and an error is calculated and fed back to the system. Then the system adjusts its parameters using this error message. The process iterates until a certain criterion is reached.

1.3 Survey of Existing Segmentation Techniques

Vidal and Marzal (1990) give a unified description of the most current available algorithmic techniques for the segmentation of speech signals. They classify the segmentation algorithms into two major categories.

1.3.1 Two Categories in Speech Segmentation

A basic algorithmic technique for segmentation of speech signals includes *unconstrained acoustic segmentation* based on spectral change and does not require any explicit linguistic category information. This technique makes use of scale-space, multilevel techniques, temporal decomposition, and Maximum-likelihood segmentation. In these methods, no linguistic categories are assumed. The system relies only on the acoustic information of the speech signal. Hence, the segments are obtained through acoustic cues and are not constrained by any categories. *Constrained segmentation* methods assume that the segments to be obtained belong to a certain set of acoustic or linguistic categories. These categories are usually chosen to model the input sequence, whereas the output units -- the resulting segments -- must be associated to these categories.

1.3.2 Unconstrained Acoustic Segmentation

Consider an example of unconstrained acoustic segmentation, spectral change segmentation, which, as the name indicates, sets the boundaries at places where it detects a large change in the magnitude of the derivative of spectral parameter vector that corresponds to a peak change of the spectral features.

Another example is the multi-level representation segmentation, or dendrogram. The speech signals are first decomposed into all the possible units in a uniform hierarchical structure. The segmentation involves a hierarchical clustering procedure which creates coarser groups until some break points are reached. These break points are the resulting boundaries.

Eberman and Goldenthal (1996) used time-based clustering for phonetic segmentation. It can be briefly summarized as follows. A sequence $\{y_i\}$ of n observation vectors is divided into an initial collection of observation blocks. This sequence of observation blocks forms the initial cluster sequence $\{C_i\}$. The distance between two clusters $d(C_i, C_j)$ is calculated using sufficient statistics computed from the underlying collection of observation vectors. For each clustering step, the smallest distance between two neighboring clusters is

selected. Clustering proceeds until the minimum distance between two clusters is greater than a threshold T . Then, these two clusters will not be further clustered and a boundary is set up between them.

As we can see from the above applications, a major property of unconstrained segmentation is that the only information it is making use of is the acoustic signal itself. The method used is a distortion measure, which is a pattern comparison algorithm that measures the dissimilarity between two feature vectors. The resulting segment boundaries do not necessarily correspond to any linguistic categories but to the abrupt change of the spectrum only.

Li and Gibson (1996) introduced a method of parametric filtering. In their method, segmentation is done by directly detecting spectral changes in the speech signal using filtering techniques in signal processing. Given two frames of speech signals, $X_t(1)$ and $X_t(2)$, the method is to derive distortion measures to quantify the deviation of $X_t(1)$ and $X_t(2)$ in their correlation structures. In doing so, a new characterization function is established. This new characterization function for representing the correlation structure of $\{X_t\}$, a real-valued stationary signal, is a certain set of output statistics from a properly designed filter bank.

Then segmentation is done by a peak-picking approach, in which a peak in the distortion is considered significant if its magnitude exceeds the threshold, and at these places we can locate the segment boundaries.

1.3.3 Constrained Segmentation

Constrained segmentation introduces additional constraints into the segmentation process. Here explicit information is used in the form of reference templates corresponding to the phonetic units. There are two types of constrained segmentation with respect to the different types of additional constraints on the segmentation algorithm. The acoustic-unit constrained segmentation has constraints on the number of acoustic units and the set of models for these units. The procedure seeks for the minimum distortion between the segments and the models.

In this procedure, the segmentation is first done arbitrarily. Then the procedure seeks for a set of models with minimum distortion for the current segments. The segmentation is done through a multistage dynamic programming technique by means of locally minimizing the distortion between the segmentation and the models. The whole procedure is iterated to refine the results.

Another type of constrained segmentation is linguistically constrained segmentation. Linguistic information is introduced in the procedure. The segments to be obtained must be consistent with the linguistic information (Vidal and Mazal, 1990: 49). The procedure uses linguistic categories as the constraints and the segments obtained must have a one-to-one correspondence to these linguistic labels.

The mechanism of this algorithm is the same as the acoustic-unit constrained method. The goal is to find the segmentation units and a set of unit models so that the distortion between them is the smallest. Most of these techniques differ only in the kind of adopted linguistic category and/or in the type of model that is assumed for these categories (Vidal and Mazal, 1990: 49).

The whole procedure is an optimization process. The input sequence is initially segmented and the distortion between the segmented sequence and the model is computed. The model is updated through an iterative procedure until the distortion reaches a small enough value.

In Sevensen and Soong (1987), a method called template matching is introduced. In this method, the phonetic transcription of the input sequence is known *a priori*. The segmentation is based on a phoneme template inventory of forty-one phonemes of American English. As in Figure 2, The

observation sequence is represented by a T speech frames. They are to be segmented into m consecutive segments. The segmentation is done through a dynamic programming procedure to minimize: $D = \sum_m \sum_n d(n)$, in which m is the number of boundaries. d is the distortion between input spectrum of reference templates. The procedure is to match an unmarked utterance with a known phonetic transcription. This transcription determines the reference templates to be used for matching. Through dynamic programming, a set of m boundaries are found for the sequence.

In a similar method used by Leung and Zue (1984), the speech signal is first segmented into broad classes using traditional statistical pattern techniques to determine relatively robust acoustic units. Then an automatic alignment of phonetic transcriptions to these units is conducted. Further segmentation based on the transcriptions is required when the matching is not in a one-to-one correspondence. In this case the segmentation has to be conducted using linguistic category knowledge.

Segment:

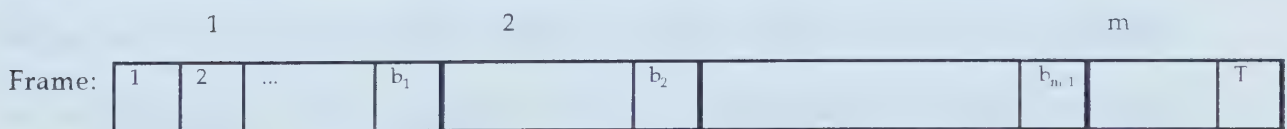


Figure 2 Segmentation template matching
(Svendsent & Soong, 1987: 77)

1.4 Segmentation Techniques

1.4.1 Model-free Method

Implementation of segmentation algorithms can be categorized into two types according to the existence or non-existence of certain kinds of models for the resulting segments. Model-free methods, or nonparametric methods, make use of a spectral distortion measure to detect change in the speech spectrum. As an example, in Deshayes and Picard (1986), the Kolmogorov-Smirnov spectral distance $D = \sup |F_1(\omega) - F_2(\omega)|$, where $F(\omega)$ is the spectral distribution function, F_1 and F_2 are taken in two frames before and after time t . If the distance exceeds a certain value, we can assume that there is a change at time t . An ideal distortion measure should be sensitive enough to detect these changes.

In actual use, the model-free method does not require any training stage for *a priori* knowledge. There is certain type of *a priori* knowledge embedded in the whole procedure which is not obtained through training, embodied in the spectral distortion measure of threshold. This method is usually used with the unconstrained segmentation algorithm.

1.4.2 Model-tied Method

The model-tied method uses certain models or templates for the resulting segments or segment categories. Each segment is associated with some kind of model. These models are usually pre-defined acoustic or linguistic categories. The problem is to find a segmentation and a model, so that the distortion between the segment and the model is the smallest.

The model-tied algorithm usually has two steps. The first step is the training session which uses some properly pre-segmented data to initialize the model parameters. This is a process of template selection. Its purpose is to find the best model for the segmentation system. The second step is the test session that involves optimal segmentation of the test data. The result of this session is the set of segmentation boundaries.

The segmentation technique based on HMM makes use of the model-tied method. The states of the HMM are taken as the segmentation units. They can be syllable-size, phoneme-size, or sub-phoneme-size. The model has to be trained to obtain the optimal model parameter. Then a segmentation procedure is conducted by finding the best state sequence through Viterbi Search.

The design of my research is an automatic procedure that makes use of appropriate acoustic categories and segments the acoustic sequence into units corresponding to these categories. The basic problem for this research can be stated as : “ Given an acoustic CVC sequence, and a set of models to model the sequence, find a set of segment boundaries for the sequence so that the resulting segments have the greatest likelihood compared to their corresponding models. The research is based on the assumption that the sequence consists of the ‘physical reality’ of smaller units and there exists a model for appropriate is modeling the segment categories.

2. Continuously Variable Duration Hidden (Semi-) Markov Model

2.1 HMM Used for Modeling Speech

Most of the current speech segmentation techniques are based on modeling each phonetic unit with hidden Markov models. They have been used as the acoustic modeling component in the most successful speech recognition systems for continuous speech recognition.

The Markov chain describes a stochastic process in which each observation belongs to a finite set of states, any observation depends only on

the immediately preceding observation, and hence has nothing to do with any other previous observations.

The hidden Markov model is a doubly embedded stochastic process with an underlying stochastic process that is not directly observable (hidden) -- the observation is a probabilistic function of the state.

We use this non-observable or 'hidden' process to model the temporal structure of the speech unit. The observation process links the acoustic vectors extracted from the speech signal to the states of the hidden process. It represents the short-term acoustic characteristics of the speech signal. An observation distribution is attached to each state.

In Huang, Ariki and Jack (1989: 80), there is a description of this process:

The HMM uses a Markov chain to model the changing statistical characteristics that exist in the actual observations of speech signals. The Hidden Markov process is therefore a double stochastic process in which there is an unobservable Markov chain defined by a state transition matrix, where each state of the Markov chain is associated with either a discrete output probability distribution (discrete HMM) or a continuous output probability density function (continuous HMM). The double stochastic processes enable modeling of not only acoustic phenomena, but also time scale distances.

Therefore, the basic idea of HMM for modeling speech is to use a fixed number of microsegments as the states of the model and the phonotactic and "microphonotactic" constraints of the sequence as the state transitions.

2.2.1 Discrete HMM

Basically, there are three types of Markov models according to the way they model the observation sequence. A discrete HMM is a discrete time, discrete state model. In this model each acoustic vector belonging either to the training or to the test set needs to be quantized using a code book. The model uses discrete output probability distributions to model various acoustic events with any distribution under the condition that the data for the training session should be large enough to exhaust all possible events.

However, the discrete model is not the best candidate for modeling a speech signal being continuous in nature. The acoustic feature space is partitioned into discrete regions by some distortion measure and the probability distributions of the original data are totally ignored. Therefore, quantization errors are inevitable. Huang (1989) mentions a smoothing technique to solve the problems caused by quantization errors. The technique includes using multiple VQ codebooks to partition parameters into separate codebooks. The advantage of the discrete model is that theoretically it could model events with any type of distribution given enough training data.

2.2.2 Continuous HMM

Another type of model, continuous HMM, has both continuous time and states. The observation does not come from a finite set but consists of a set of continuous points. Therefore, the output distribution should take the form of continuous output probability density functions. It models the acoustic observation directly using estimated continuous probability density functions without vector quantization.

Hence the observations need to be described in a proper form of probability density function. In the use of continuous probability density functions, the first candidate for a family of output distributions is the family of multivariate Gaussians, since:

1) Gaussian mixture densities (with an appropriate chosen mixture) can be used to approximate any continuous probability density function in the sense of minimizing the error between two density functions.

2) by the central limit theorem, the distribution of the sum of a large number of independent random variables tends towards a Gaussian distribution.

3) the Gaussian distribution has the greatest entropy of any distribution with a given variance. (Huang, 1989: 49)

The advantage of the continuous model is that the continuous observations can be modeled directly without quantization and therefore the quantization errors can be avoided. The most general representation of the pdf, for which a reestimation procedure has been formulated, is a finite mixture of the form

$$b_j(\mathbf{o}) = \sum_{k=1 \dots M} c_{jk} N(\mathbf{o}, \mu_{jk}, \mathbf{U}_{jk}), \quad 1 \leq j \leq N$$

where \mathbf{o} is the observation vector being modeled, c_{jk} is the mixture coefficient for the k th mixture in state j and N is any log-concave or elliptically symmetric density (e.g., Gaussian) (Rabiner & Juang, 1993: 350). And for explicitly modeling the parameter correlations, it is used with full covariances. However, this increases greatly the computational complexities. Therefore, many models use mixture Gaussian with diagonal covariances, which completely ignores the parameter correlations (Ljolje, 1994).

An alternative is to use a single multivariate Gaussian distribution with a full covariance matrix. We adopt this method in the experiment based on Ljolje (1994) that a single full covariance Gaussian distribution which

explicitly models the parameter correlations performs much better than weighted mixtures of Gaussian densities with diagonal covariances which implicitly model the parameter correlations.

2.2.3 Semi-continuous HMM

As a compromise between the discrete and continuous models, semi-continuous HMM provides a good solution to the conflict between detailed acoustic modeling and insufficient training data. In this model, there is still a VQ codebook. The VQ codebook consists of a mixture of continuous probability density functions but each codeword of the codebook is represented by one of the probability density functions. These probability density functions are overlapped rather than separated. So, for modeling the acoustic event, one of the probability density functions may be used with others.

2.2.3.1. State Durations

In addition, a semi-Markov chain takes into consideration the state durations. Guedon (1996) describes the mechanism of a semi-Markov chain: “when a discrete semi-Markov chain enters a state, it determines the next state to which it will move according to the transition probabilities of an ‘underlying’ Markov chain. After the next state has been selected, but before

making any transition, the process holds for a random time in the current state according to its occupancy distribution" (Guedon, 1996: 137).

State durations are described either by durational probability density functions or by some lower-order statistics. There are two ways of modeling durational distributions, either implicitly or explicitly. In the standard hidden Markov model, durational distribution is described by implicit state durational probability density functions. Implicit state duration refers to the number of consecutive observations (duration) in one state, i.e., the number of consecutive self-transitions. Therefore, if the system is in a known state i , the probability that it stays in that state for d consecutive observations is given by an exponential function:

$$p_i(d) = (a_{ii})^{d-1} (1-a_{ii})$$

where a_{ii} is the transitional probability from state i to i -- self-transition coefficient.

However, for real speech signals, this geometric description is generally inappropriate. It is more reasonable that we model them explicitly rather than implicitly, base on the process they undergo, which means we would rather describe the speech signals as staying in one state for a certain period of time. In this case, an explicit duration density is specified and there is no self-

transition any more. This is the property of a semi-Markov model. Figure 3 shows the HMM with and without explicit durations. We can see that self loops in the HMM no longer exist in semi-HMM. In Ljolje and Levinson (1991), an experiment is done with a semi-HMM, followed by an identical recognition experiment using the traditional form of a HMM which does not have an explicit duration model but preserving the rest of the model parameters. The absence of a correct duration model increases the error rate by 50%. According to Ljolje and Levinson (1991: 29), "Explicit duration models, even if only specifying the longest and shortest durations allowed for a speech segment, can be beneficial to the recognizer performance." This is the method we adopted in this research.

	without duration constraint	with duration constraint
recognition	80.61%	86.83%
segmentation	83.48%	84.48%

Table 1 Results of Wang's experiment

Certain researches in speech recognition and segmentation have been done to deal with both the implicit and the explicit durational distributions. For the implicit durations, the method is to train the model with constraints on the durational statistics of the acoustic segment. Wang (1994) investigated

the durational behaviour of the HMM by durationally constrained training of HMM with implicit state durational pdf. His results are shown in Table 1.

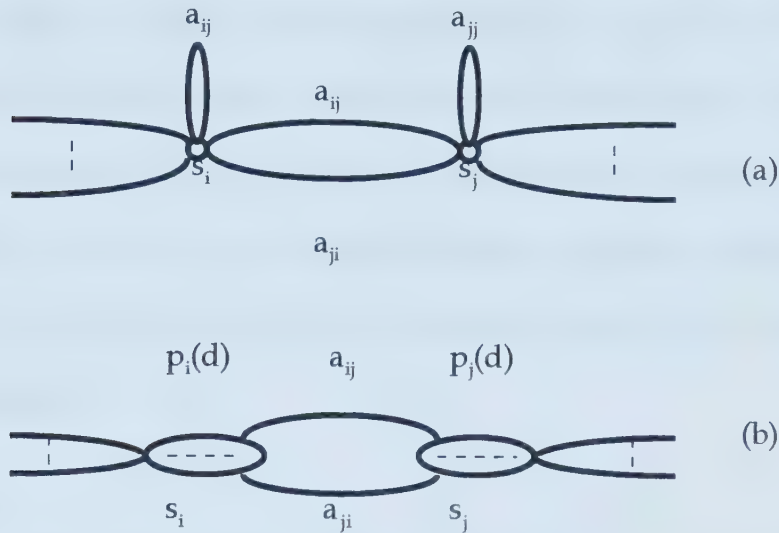


Figure 3 Illustration of general interstate connection of (a) a normal HMM with exponential state duration density, and (b) a variable duration HMM with specified state densities and no self-transitions from a state back to itself. (Rabiner & Juang, 1993: 358)

As may be seen, there is improvement in both recognition and segmentation. For recognition, the accuracy rate increases from 80.61% to 86.63%. But the improvement in segmentation is not very apparent, only by 1.00%.

Because of their nature, explicit state durations are modeled by variable duration models. The performance of the HMM can be improved using variable duration distributions. However, there are some drawbacks using

these kind of distributions. In these models, the parameters in the distribution function are obtained in the training session, which means there are more parameters to be trained besides the usual HMM parameters. This will result in a great increase in the computational complexity. Therefore, there is proposal to use parametric state duration distributions. There are certain types of distributions used to describe explicit state durations. Russell and Moore (1985) proposed to use the Poisson state occupancy distribution for a semi-Markov chain. Levinson (1986) investigated semi-Markov chains with gamma distributions.

Gamma distribution takes the form of

$$d_j(t) = \frac{\eta_j^\gamma}{\Gamma(\gamma)} t^{\gamma-1} e^{-\eta_j t}.$$

Guedon (1992: 380) describes the reason of adopting Gamma distributions, since “Gamma distributions present a satisfying compromise between the amount of data necessary for a confident estimation and the modeling capabilities. With only two parameters, it is possible to model exponential distributions (for $\gamma=1$) as well as normal distributions (for high γ) and all intermediates between these two extremes”. The implementation of the Gamma distribution is to find the mean duration of each phoneme and its variance for each of the states, which are then converted into the parameter γ and η .

Another possibility, which has been used with good success, is to assume a uniform duration distribution over an appropriate range of durations, and use a path-constrained Viterbi decoding procedure (Rabiner & Juang, 1993: 362), which is used in the implementation of this research. The constraints are also set for the minimum and maximum durations of each microsegment.

3. A Statement of the Research

This paper deals with a segmentation algorithm for parsing speech signals into units smaller than a phonological segment, which we call microsegments. The purpose of the study is to develop a robust and reliable autosegmentation algorithm.

To fulfill this goal, the first step will be concerned with the training session, whose task is to train HMM-like structure for optimal parameters. The data for the training session are a set of hand-marked CVC patterns. The hand-marked cursor marks the boundaries of the microsegments, and these microsegments are chosen as the states of the model. The tasks in this stage are to set up a model $\lambda=(A, B, \pi)$ first and then train the model to give each state an 'identification' by using feature vectors consisting of mel-cepstral

coefficients. The purpose of doing so is to create the best models for the real phenomena.

The next step in the procedure is to find the best matching state sequence. The data will be put through the model, and once the observation sequence $O=(O_1,O_2,...O_T)$ is obtained, the corresponding state sequence $q=(q_1,q_2,...q_T)$ has to be chosen that is optimal in some sense. The method adopted will be the Viterbi algorithm, and the ultimate objective is to put cursors at these boundaries.

CHAPTER 2 DESCRIPTION OF THE IMPLEMENTATION PROCEDURE

Chapter two gives the theoretical background of the development of the model. Implementation of the whole procedure is also introduced in this chapter.

1. Introduction

1.1 Microsegments Used in This Research

The motivation of this autosegmentation system is based on the assumption that a phoneme-sized segment can be further represented by smaller units, here referred to as microsegments. Since different phonemes tend to have similar subphoneme segments (such as bursts in stops), it is reasonable to introduce a lower level description of phonemes. At this level, a phoneme is further divided into smaller units that show roughly uniform properties. The microsegments used in this research are the same as those used in Nearey (1992): voice bar, burst, vowel onset, and vowel end. According to Nearey (1992: 4), these microsegments can be defined as:

voice bar: a relatively low amplitude, quasi-sinusoidal waveform that precedes the release burst of some voiced stops.

burst: a brief transient followed by noise that marks the opening of the vocal tract, the so called “explosion phase” of a stop consonant. This may be followed by a brief frication and by aspiration. No effort was made to delineate these events.

vowel onset: the onset of steady voicing associated with the vocalic part of a syllable.

vowel end: the low energy “tail” of a vocalic (vowel-like) chunk preceding a final consonantal chunk in the CVC’s.

Similar microsegmental units have served as the basis for numerous descriptive studies of speech. In Deng, Lennig, and Mermelstein (1990), subphonemic level units for stops are proposed to model the states of the HMM to improve the performance of a 75000-word speaker-dependent recognition system. Their research studied the problem of poor discrimination of stop consonants in a phonemic model (each state of the model is used to model a phoneme). The microsegments they used for stops are: silence, voice bar, burst, and aspiration. In addition, context information

in the form of frontness of the following vowel is introduced for modeling burst and aspiration. Their experiments show that with the use of microsegments models, the error rate is reduced by 35% compared with the results from the phonemic HMM (Deng, et. al, 1990: 2738).

The purpose of this segmentation system is to automatically set the boundaries for the above microsegments. In fulfilling this task, a model will be set up for the system, using the microsegments as its states. This is based on the assumption that each microsegment contains significant acoustic information for the discrimination of the unit from other such units. The statistics for each state are obtained through the training data which are manually marked at these boundaries. Then, a Viterbi search is conducted to relate the observation sequence to the best state sequence. In some models, some additional “pseudo microsegments” which are not explicitly marked by observers are introduced, as will be described below.

2. The Implementation Procedure

2.1 Theoretical Basis for the General Model

A Hidden Markov Model can be thoroughly described by its model parameters $\lambda=(\pi, A, B)$, in which A is the transition probability matrix, B is the observation distribution function, and π is the initial state probability.

The method adopted in this research does not quite follow the procedure of the general application of HMM. It simplifies the model in certain ways for implementation purposes. In this sense, it is a “relaxed” version of a traditional HMM.

According to Huang (1989), there are three basic problems of interest (basic steps) in the applications of HMM. The first problem is the selection among several competing models for the best one to model the observation sequence. A solution to this problem is the evaluation of $P(O/\lambda)$ by the enumeration of every possible state sequence of the length of the number of observation. In real application, the backward or forward procedure is used to find out a model with the highest $P(O/\lambda)$. In our research, this problem does not exist because only one model is adopted for all CVC sequences.

Another problem is that given a model, how do we adjust the parameters to maximize its output? A solution to this problem is to conduct the parametric estimation. The most commonly used parametric estimation is the Baum-Welch method, known as the EM (expectation-maximization). This is a procedure to maximize the current parameters of a given model locally with the obtained observations and to use the new parameters to replace the original model parameters and then re-estimate them with new observations. The procedure is iterated until a certain criterion is reached. In

the current research, we do not use this procedure either. Rather, we assume that in the training data, the state sequence is known rather than hidden and that we know the times of onset and duration of each state.

The third problem is that given the observation sequence and the model, how do we choose a corresponding state sequence that is optimal in some sense? This research is directed to the solution of this problem. The first step for approaching the problem is to give a definition of what an optimal state sequence is.

There are several ways of choosing the optimal state sequence. One is to choose the states that are individually optimal, i.e., each individual state has the highest probability. Another method is the Viterbi algorithm. The Viterbi algorithm is to find the single best state sequence, i.e., to find a path with each step along the path having the best score.

2.2 Certain Modifications Adopted in This Research

In this research, certain of the model parameters are chosen *a priori*. Since we can assume that the speech signal always starts with silence, the initial state probability for silence is set to one, and the rest to zero. The state transitions are also set *a priori*. They are the reciprocal of the total number of allowable paths out of each state. The allowable transitions from state to state

are determined by theoretical constraints or “microphonotactic” constraints. For example, there are two possible transitions for the silence, either to the voice bar or to the burst (Figure 6 presented later in this chapter gives the full description of possible transitions).

The observation distributions are modeled by a “single mixture” (i.e., simple, unimodel) multivariate Gaussian density function and the state duration is described by uniform distribution for speed in implementation. This model, in a strict sense, is not a standard hidden Markov model because it does not involve the procedures for obtaining the optimal state parameters in the training, rather the “correct” state sequence is assumed to be known *a priori* for the training set. However, a Viterbi search is conducted for finding the best path, in a way very similar to that used in HMM procedures. Therefore, this model can be regarded as a modified or a relaxed assumption hidden Markov model. A Viterbi algorithm finds the best sequence by locally optimizing each step of the search. In this sense, this research procedure can also be described as an dynamic programming procedure.

3. Data

3.1 Training and Testing Data

Training data are used for calculating the statistics of the states in the model, i.e., each state is represented by a feature vector from the training session. This feature vector representation is later used as a template for the automatic segmentation system. Therefore, these statistics are directly related to the performance of the model because they determine the output results. For the model to act accurately in most situations, the data should be rich in both contexts and speakers. The sample size should be large enough so that it would contain all the information that is necessary for making the right decision.

Testing data are used for evaluation of the model performance. Therefore, the data should include reference points for analysis of the output results, i.e., the testing data must include hand-marked cursors.

In this research, data for training and testing are called BSET, referred to as the calibration training set, or the benchmark set. The BSET data consist of 720 recorded tokens of CVC patterns from twelve speakers, five male and

seven female. The initial consonant contains six initial stops /p, t, k, b, d, g/, and there is only one final consonant /k/. Ten vowels in Canadian English are used, which are /i, ɪ, e, ɛ, æ, ʌ, ɒ, o, ʊ, u/.

The data are carefully hand-marked by phonetically trained linguistics students to put cursors at the microsegment boundaries of their waveforms. These microsegments, as stated previously, are: 1) voice bar, 2) burst, 3) vowel rise, and 4) vowel end. These microsegments are relatively homogeneous chunks of acoustic signals smaller than a phoneme. They maintain a relatively high degree of homogeneity within the hand-marked boundaries and correspond to certain relatively well-defined production properties of speech sounds (Fant, 1970).

In the production procedure, a stop involves a vocal tract closure associated with acoustic silence. In the situation when voicing is present, low frequency energy will appear. This low frequency energy characterizes the voice bar. The vocal tract closure is followed by an abrupt release of the constriction which produces a kind of transient noise. This transient noise is called the burst. The 'transient' itself is usually very short (less than 10ms), after which there is often a brief period of fricative and/or aspiration. It is usually followed by a transition from the stop to another sound (a vowel for instance) (Kent and Read: 1994).

The cursors are placed at the following microsegment boundaries: 1) initial consonant silence, 2) start of voice bar, 3) start of burst, 4) vowel rise, and 5) vowel end (Figure 4).

Marks for the events associated final consonant were not available in these data. However, as noted by Nearey and Kiefte (1994), the events from the final consonant resemble those of the initial stops. Therefore, micro-segmentation of the consonant can be based on the information from the initial consonant. Observation distributions for the events associated with final /k/ can be constructed on the basis of statistics from the initial stops. This is an example of exploiting the reusable nature of micro-segmental models.

The hand-marked cursors are very critical for the entire algorithm in that they are reference points for feature extraction and also boundaries of the states of the HMM. In addition, they are further used as a reference for the evaluation of the automatic segmentations, since the manual segmentation and labeling of speech, although subject to human inaccuracy, rarely results in gross errors of judgment. Hence, automatic segmentation will be considered to achieve better results if the boundaries are set closer to the corresponding hand-marked cursors in a test data set.

4. Extracting Feature Vectors

4.1 “Standard” Feature Representations

Some parametric speech analysis is necessary to characterize the speech signal for modeling purposes. Once the signal is processed, it is represented by a sequence of parameter vectors. Acoustic feature vectors often take the form of k -dimensional parameter vectors, representing the speech signal within an analysis window. These are based on short-time speech analysis techniques of either frequency or time domain. “Short-time” here refers to the windowing techniques used in speech processing, where speech waveforms are truncated by window functions for analysis purposes. This technique is based on the fact that in the long run speech signal is non-stationary, but within a certain period of time it is relatively stable. Within the frequency domain, filter banks, FFT, and cepstral analysis are commonly used. Time-domain measures make use of autocorrelation functions, zero-crossing rate, and signal energy.

Cepstral analysis is a fairly standard method for speech processing. Cepstral coefficients are coefficients of the Fourier transform representation of the log magnitude spectrum. Acoustic information is analyzed into frequency groups. Two types of cepstral coefficients are used: mel-cepstral coefficients represent the acoustic information within the frame, and delta-cepstral

coefficients describe the changing in mel-cepstral coefficients over a sequence of frame.

4.2 The Calculation of Feature Vectors in This Research

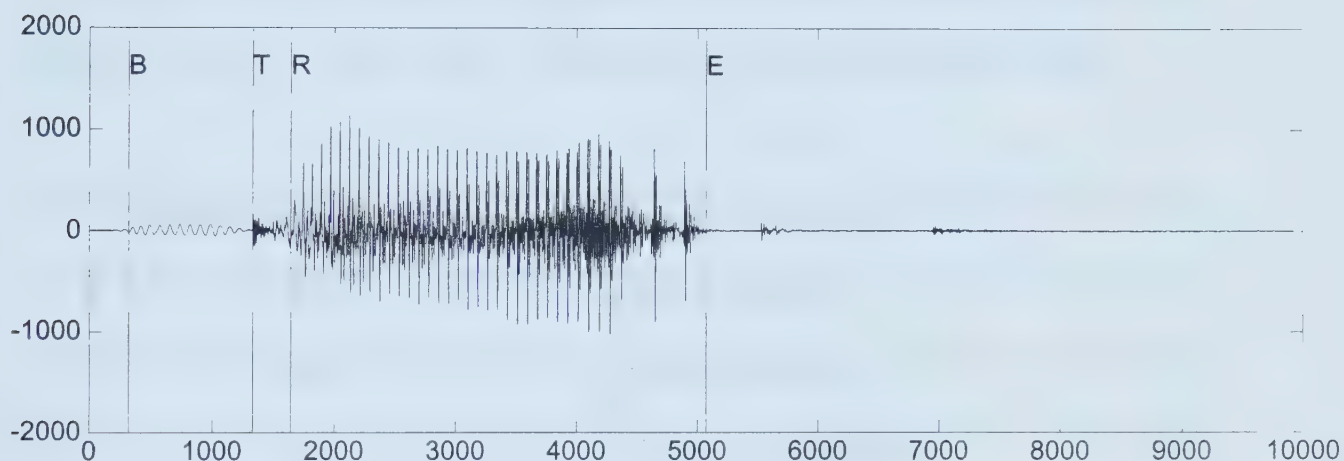


Figure 4 Signal with hand-marked cursors (B -- voice bar, T -- burst, R -- vowel rise, and E -- vowel end).

The first experiment uses mel-cepstral coefficients to represent the acoustic information. The number of the coefficients is varied in the third experiment to determine a “best” number for an adequate representation of speech for the purpose of microsegmentation. Later, in experiment four, delta cepstral coefficients are added to try to capture certain additional dynamic aspects of the signal. In calculating the feature vectors, the signals are sampled at the rate of 16 kHz. The sampled waveform is windowed by a 20 ms Hamming window with a 5 ms frame advance. An FFT is performed on each

window with a size of 512 samples. Spectral analysis is performed, using a filter bank over FFT samples.

As mentioned above, the speech signals are hand-marked by students with phonetic knowledge. Table 2 gives the hand-marked cursors. The purpose of hand-marking the segments is to provide reference points for calculating feature vectors and a set of preliminary states for the model.

Hand-marked cursors	Cursor positions
M1	initial silence
M2	initial voice bar
M3	initial burst release
M4	vowel start
M5	vowel end

Table 2 Hand-marked cursors and their descriptions

The feature vectors for each frame consist of thirteen mel-cepstral coefficients in Experiment 1. A second analysis of nine coefficients are also used to see what results can be obtained in Experiment 2. Later delta-cepstral coefficients are calculated and added to describe the dynamic properties of the speech signal. This is done in experiment four. At this time, 13 mel and 13 delta cepstral coefficients are used, 26 parameters in total.

The spectrum was represented using cepstral coefficients, where the relationship between the spectrum magnitude and the resulting cepstral coefficients is defined by:

$$C_i = \sum_{j=1, \dots, N} \text{Log}(E_j) \cos((j-1/2)\pi/N), \quad i = 1, \dots, N-1$$

Here N is number of filters and E_i is log energies of each band.

And delta cepstral coefficients are calculated using the formula as the first order linear regression coefficients:

$$d_t = \sum_{\tau=1, \dots, N} \tau (c_{t+\tau} - c_{t-\tau}) / 2 \sum_{\tau=1, \dots, N} \tau^2$$

Here c_t is the cepstral coefficient at time t . At the edges of the data, the following simple first order differences are used:

$$d_t = c_{t+1} - c_t, \quad t < N$$

4.2.1 Continuation Part of the Microsegment

Owing to the different configurations in the production mechanism, there is reason to assume that the behavior of the onset part of the microsegments is different from its continuation part. The onset part presents an abrupt change and the continuation part is relatively steady. Therefore, the statistics taken from this part may be expected to differ from those of the continuation part. It is more reasonable to use different feature vectors to represent these parts. A better result may be achieved if we model them separately. Thus the voice bar is further divided into voice bar onset and voice bar continuation part, and the burst into burst onset and burst continuation, each associated with different distributions. The vowel has three distributions: vowel onset, vowel continuation and vowel end. The statistics for the continuation part are taken from the middle of each microsegment to avoid the influence of the onset and offset of the segment.

As mentioned above, the onset and offset are already hand-marked with cursors. The continuation part is further marked at the center of each microsegment through programs. Table 3 gives the basic signal types defined by feature vectors. There are two kinds of distributions associated with the voice bar and burst of consonant respectively. The vowel part has three distributions: vowel onset, vowel continuation, and vowel end. The onset

parts are labeled *a* and the continuation parts *b*. The end of the vowel is labeled *c*. Figure 5 shows the location of the onset versus continuation part of the signal (The capital letters represent the onsets, *B* -- voice bar, *T* -- burst, *R* -- vowel rise, and *E* -- vowel end. The small letters represent the offsets of each microsegment, *v*, *g*, and *h* represent the continuation parts of voice bar, burst, and vowel respectively).

For the onset parts, feature vectors are calculated on the frames one before and one after the hand-marked cursor. For the continuation parts, the first frame is set to start from the center of the microsegment, and the feature vectors are calculated out of the frames starting from the first frame until the last frame before the next microsegment starts. The onset and offset portions of the waveforms (the type *a* and type *c* often look different from the portions from the center of a segment (type *b*). In addition, it seems likely that changes in the cepstral coefficients over time will show different patterns near the edge portions of the microsegments than in the continuation parts and this difference in patterning might be exploited by the delta cepstral coefficients described later in this chapter. To avoid influences of onsets and offsets, continuation parts are trained only on tokens with relevant microsegment durations of at least 30 ms.

5. Transitions of the Model States

The HMM used for speech recognition, at the acoustic level, consists of

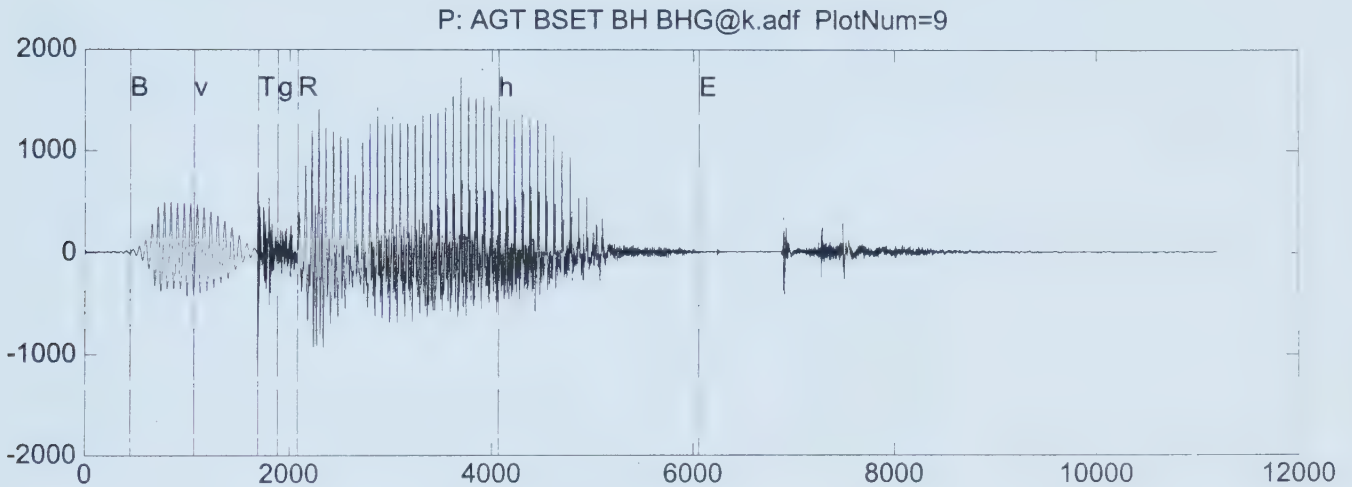


Figure 5 Locations of onset *vs* continuation part of the signal

a set of states and an associated transition process. In this research, for ease of implementation, the state-transition probabilities a_{ij} are set *a priori*. The probability of each exit path is taken as the reciprocal of the total number of such paths for that state, i.e., all exit paths from one state have equal chances to be taken. As an example, for a state of four exit paths, there is a 25% chance of taking each path.

5. Transitions of the Model States

The HMM used for speech recognition, at the acoustic level, consists of

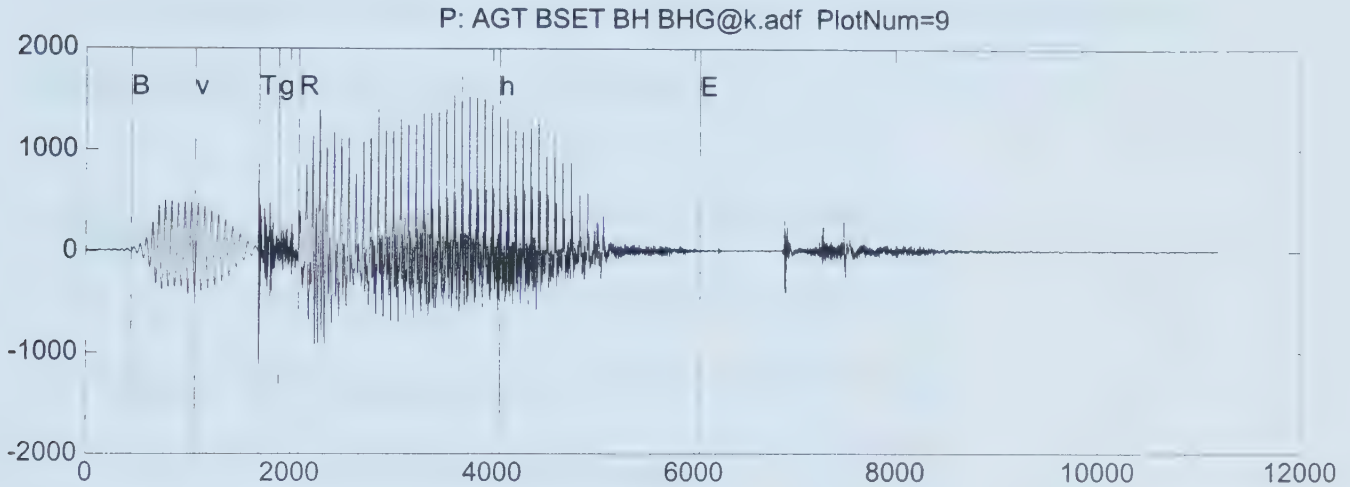


Figure 5 Locations of onset *vs* Continuation part of the signal

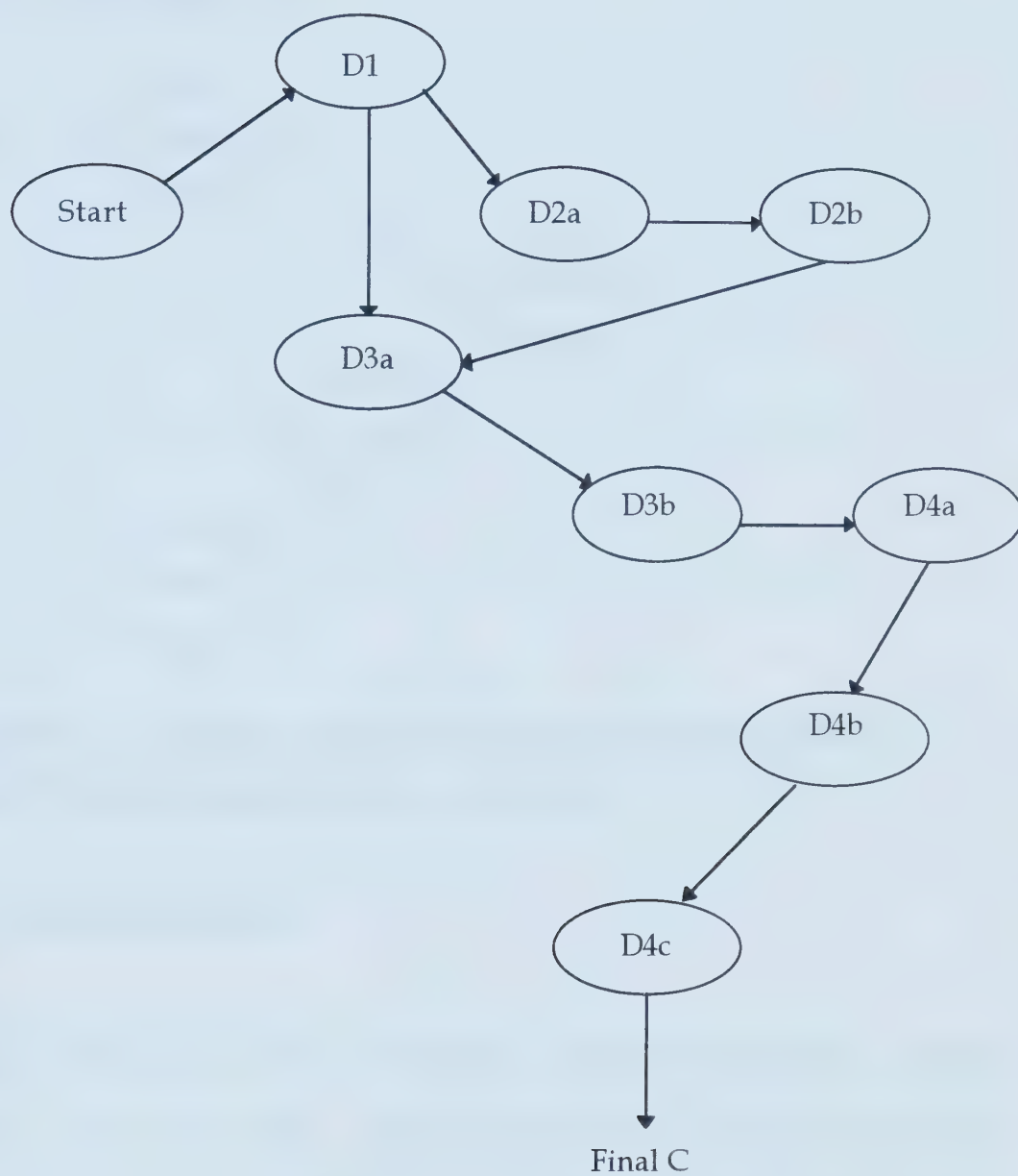
a set of states and an associated transition process. In this research, for ease of implementation, the state-transition probabilities a_{ij} are set *a priori*. The probability of each exit path is taken as the reciprocal of the total number of such paths for that state, i.e., all exit paths from one state have equal chances to be taken. As an example, for a state of four exit paths, there is a 25% chance of taking each path.

Figure 6 presents the states and their possible transitions based on *a priori* speech knowledge. The transitions take place as indicated by the arrows. Hence, in the Viterbi algorithms, the final scores always correspond to the alignment of the observation sequence with the best state sequence ending in the final state. In addition, the state durations are modeled by uniform distributions as described later in this chapter.

Signal Type	Description
D1	silence
D2a	voice bar onset
D2b	voice bar continuation
D3a	C1 burst onset
D3b	C1 burst continuation
D4a	vowel onset
D4b	vowel continuation
D4c	vowel end

Table 3 Basic signal types defined by feature vectors.

Figure 6 Possible state transitions



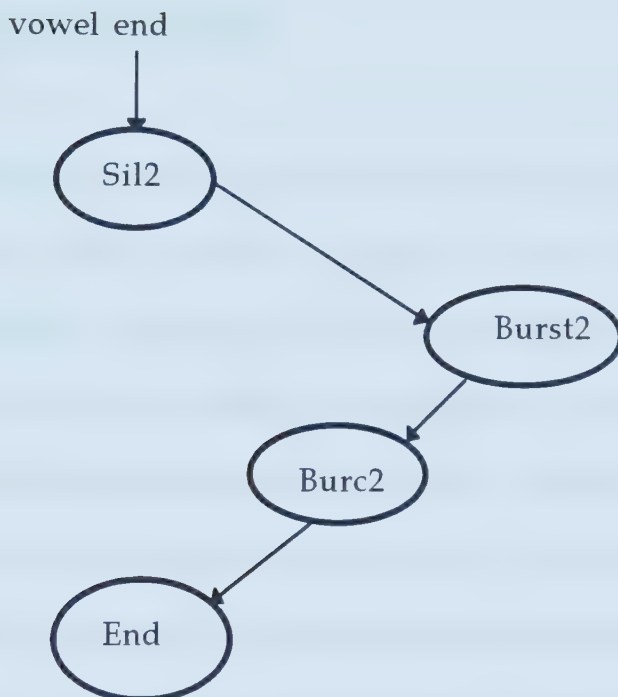


Figure 7 Possible transition of states for final segments (Sil2 -- silence of the final stop, Burst2 -- burst onset of the final stop, and Burc2 -- burst continuation of the final stop)

6. Modeling of State Duration

For modeling the duration distribution, this research uses the method in Rabiner and Juang (1993), which is to assume a uniform duration probability over an appropriate range of durations, and uses a path-constrained Viterbi decoding procedure. The implementation of the Viterbi search uses a self-loop to set constraints on duration range.

7. Modeling of Final Segment

Experiment 2 tests the effect of modelling the final segment more explicitly. The final consonant is always /k/. The possible transitions are shown in Figure 7. As /k/ is a voiceless stop, the voice bar does not exist. There are several ways of dealing with the final /k/. One way is to hand-mark it, as we did before, and retrain the model as a whole. This seems to be the most reasonable or reliable way. However, this method was not adopted because the hand-marked cursors were not available for the BSET data and redoing it was deemed too time-consuming. An alternative (Nearey & Kieffe: 1994) would be to copy the statistics of the first consonant to the final set of states, assuming that they behave in a similar way. The architecture of the final segment is shown in Figure 7.

8. Observation Distributions

As mentioned above, each state of the model is described by a Gaussian probability density function. In Ljolje (1994: 231), better recognition performance and likelihood scores are achieved for modeling acoustically similar sub-word speech units which are also linguistically motivated using

explicit modeling of cepstral parameter correlations. In this research, a “single mixture” multivariate Gaussian pdf is used with full covariances.

9. Viterbi Search

A Viterbi algorithm is used to find the ‘best’ state sequence or, In other words, the state sequence which is most likely to occur coincidentally provides the best estimate of the microsegment string of the utterance. Thus, the problem here is: Given an observation sequence, how do we find an optimal state sequence to account for it. There are four steps in the complete procedure for finding the best state sequence in the standard Viterbi algorithm (Huang, 1993).

The Viterbi algorithm states that given an observation sequence $O=(O_1 O_2 \dots O_T)$, the best state sequence $q=(q_1 q_2 \dots q_T)$ is the one with the highest probability

$$\delta_t(i) = \max_{q_1 q_2 \dots q_{t-1}} P[q_1 q_2 \dots q_{t-1}, q_t = i, O_1 O_2 \dots O_t | \lambda].$$

Let $b(o_i)$ be the observation symbol probability distribution. And we use a pointer ψ to keep track of the state sequence when we get maximum δ .

Step 1. Initialization

$$\delta_1(i) = \pi_i b_i(o_1), \quad 1 \leq i \leq N$$

$$\psi_1(i) = 0$$

In this experiment, the signal always starts with silence. We can assume $\pi_1 = 1$, and the rest of initial probability distributions are all zero.

Step 2. Recursion

$$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}^*] b_j(o_t) \quad \begin{matrix} 2 \leq t \leq T \\ 1 \leq j \leq N \end{matrix}$$

$$\psi_t(j) = \arg \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}^*] b_j(o_t) \quad \begin{matrix} 2 \leq t \leq T \\ 1 \leq j \leq N \end{matrix}$$

This step states that among all the possible transitions from state i to j , we are going to find out the one with the highest score. There is an additional set of constraints relating to the probability a_{ij} of a transition to state j from state i that are involved in implementing the minimum and maximum state durations. The minimum durations of each microsegment (initial silence, voice bar onset, voice bar continuation, burst onset, burst continuation, vowel onset, vowel continuation, vowel end, silence before

final stop, final burst onset, final burst continuation, and final silence) are: 5, 2, 10, 2, 10, 2, 35, 2, 5, 5, 5, 5 ms. The maximum durations are set as the duration mean of each microsegment plus five times of its standard deviation. These constraints are given by the following equations:

$$a_{ii\tau}^* = 1, \tau < t_{\max(i)}$$

$$a_{ii\tau}^* = 0, \tau \geq t_{\max(i)}$$

$$a_{ij\tau}^* = 0, \tau < t_{\min(i)}$$

$$a_{ij\tau}^* = a_{ij} \tau \geq t_{\min(i)}$$

Here, τ is the elapsed time in state i , $t_{\max(i)}$ and $t_{\min(i)}$ are the maximum and minimum allowed durations for state i , $a_{ii\tau}^*$ is the transition “pseudo-probability” for staying in state i , and $a_{ij\tau}^*$ is the probability for transiting from state i to (the distinct) state j at duration τ . The net effect of these constraints is to force state occupancy in state i until its minimum duration is reached, to allow either continued state occupancy or a transition to another state for all legal durations, and to force exit of state i when its maximum duration is exceeded.

Step 3. Termination

$$P^* = \max_{1 \leq i \leq N} [\check{\delta}_T(i)]$$

$$q_T = \arg \max_{1 \leq i \leq N} [\check{\delta}_T(i)]$$

The final state is reached.

Step 4. State sequence backtracking

$$q_t^* = \psi_{t+1}(q_{t+1}) \quad t = T-1, T-2, \dots, 1$$

Then, we may track back from the final state to recover the optimal state sequence and thus an optimal microsegment sequence.

CHAPTER 3 SEGMENTATION EXPERIMENTS AND ANALYSIS

Four experiments with different conditions are conducted. The results and analysis are given in this chapter.

1. Introduction

The main segmentation algorithm makes use of a modified HMM. The states of the HMM are chosen as the microsegments of the speech data. For the final segments, the observations for each state are based on similar states from the initial consonant, as it is assumed that the final stops can be treated the same as the initials since they consist of the same microsegments used in the experiment. The state transition probabilities are set *a priori*. The observation distributions are modeled as single mixture multivariate Gaussian distributions of a full covariance matrix. For modeling the durational distribution, a uniform distribution is adopted. This method is adopted because it is computationally efficient and good results can be achieved when implemented in a path-constrained Viterbi decoding procedure (Rabiner & Juang, 1993: 362).

The research consists of two stages of experiments. It begins with a training phase to estimate model parameters. In this stage, the cepstral coefficients are calculated for each microsegment as its representation. After training, the testing stage involves automatic segmentation using a Viterbi search. The output is a set of cursors placed at the estimated microsegment boundaries.

Several experiments are designed to investigate the influences of certain variations in the model architecture. A series of analyses of the experiment results are presented using values derived from the differences between the hand marked cursors and the estimated ones.

2. Two Stages of Experiments

2.1 Training

Tokens chosen from BSET are used for training, which includes 345 tokens from subjects *A*, *B*, *C*, *D*, *E*, and *F* (three male and three female subjects). As mentioned above, BSET data contain carefully hand-marked cursors placed at the microsegment boundaries. These cursors are used to calculate feature vectors and identify the model state boundaries. A program

selects the time points in the signal corresponding to the nominal state boundaries.

The sampled waveform is windowed by a 20 ms Hamming window with a 5 ms frame advance. For the onset parts, feature vectors are calculated out of the frames one before and one after the hand-marked cursor, to allow for a small amount of temporal uncertainty near the boundary. For the continuation parts, the first frame is set to start from the center of the microsegment, and the feature vectors are calculated out of the frames starting from the first frame until the last frame before the next microsegment starts. Each feature vector consists of 13 mel-cepstral coefficients.

2.2 Testing

The data used for testing are the reserved half of BSET data (subjects G, H, I, J, K and L, four female and two male). The input are the CVC tokens from the data and output is a set of segment boundaries. Cepstral coefficients are calculated as in the training. Then, a Viterbi Search is conducted to find out the best state sequences and duration of each state. Cursor time is determined as the beginning time of each relevant state. The hand marked cursors are used as reference points for the result analysis.

In order to compare and improve the model performance, a series of experiments are conducted and results are analyzed. Experiment 1 is conducted on the conditions discussed in the introduction section of this chapter.

In Experiment 2, the importance of modeling final segments is illustrated by conducting an experiment without final segmentation, and comparing the results with Experiment 1.

The following experiments three and four manipulate the use of cepstral coefficients. In experiment three, the number of mel-cepstral coefficients is changed to see the effect on the results because we want to find out if the 'standard' 13 cepstral coefficients used in the experiments is the 'best' choice. Later the delta-cepstral coefficients are introduced to capture the dynamic information of the microsegments.

Data for the results analysis are taken from the values of estimated cursors subtracted by the hand-marked cursors. The analysis includes calculating the mean, root mean square errors, and lower and upper quartiles of the errors. A boxplot display is presented for each experiment to show positions of the lower quartile, median, and upper quartile values. Also the distribution of outliers is indicated.

3. Segmentation Experiments and Results

3.1 Segmentation Experiment 1

The experiment uses the method discussed above. In summary, the segmentation is conducted by a Viterbi search along the path shown in Figure 4. Each state in Figure 4 is represented by a feature vector of 13 mel-cepstral coefficients.

	mean(ms)	rms	25% (ms)	75% (ms)
B	-10.16	23.10	-26.75	1.38
T	-1.77	12.46	-9.52	4.33
V	13.12	40.79	-2.75	21.5
E	25.51	52.96	-1.41	50.44

Table 4 Results of Experiment 1

Also, based on Ljolje (1994), we assume that explicit modeling of the parameter correlation of the observation improves the model performance. Therefore, the experiment uses a full covariance matrix for the multivariate

boxplot. Starting from the left, it shows the results of voicebar, burst, vowel rise, and vowel end. The three horizontal lines of the box indicate the lower

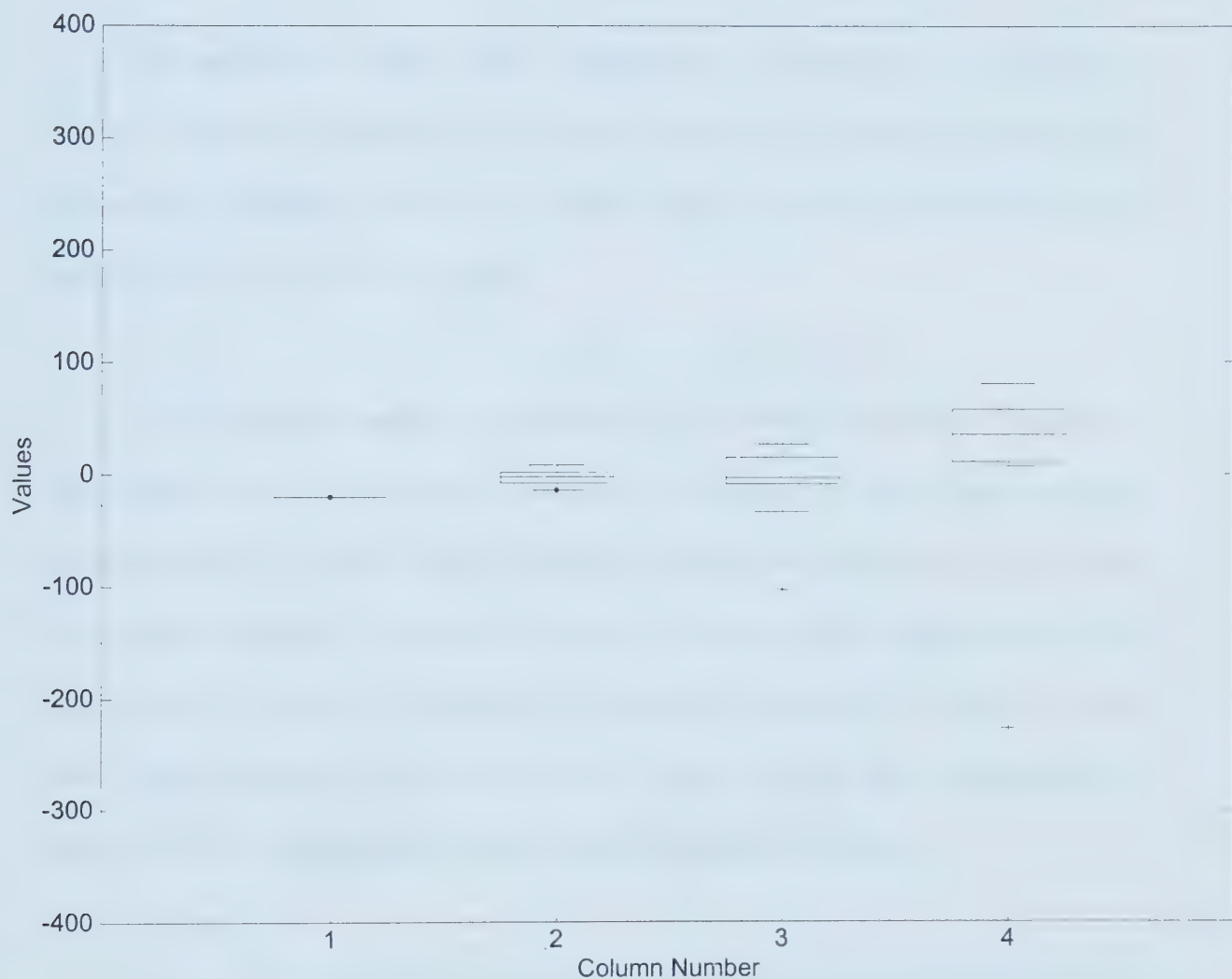


Figure 8 Boxplot of results for Experiment 1 (1-- voice bar, 2 -- burst, 3 -- vowel rise, and 4-- vowel end)

quartile, median, and upper quartile values. The whisker lines extending from each end of the box show the extent of the rest of the data. Outliers are data with values beyond the ends of the whiskers.

3.2 Experiment 2 -- Segmentation of Final Consonant

As mentioned above, the segmentation procedure is conducted by Viterbi algorithm. Because the Viterbi search is a global optimization procedure, modeling of the end of the signal can have side effects that improve the fit of the CV sections.

Hand-marked cursors do not exist for the final consonants. Therefore, the statistics from the initial consonant are copied to the final segment, assuming that they have similar properties. Then, segmentation is conducted using these statistics. The procedure still involves a Viterbi search along the path shown in Figure 7. Since final consonant is always /k/, there are only four states associated with it. They are 1) silence of the final consonant, 2) burst, 3) burst continuation, and 4) end of the final consonant.

The model used in Experiment 1 included the segmentation of the final consonant. Experiment 2 will show the performance of the model without modeling the final segment. Table 5 and Figure 8 give the results and boxplot. Compared with the results of Experiment 1, there is a big increase in

the *rms* values for all microsegments, 19.72 for voice bar, 59.16 for burst, 75.23 for vowel rise, and 204.51 for vowel end.

	mean(ms)	rms	25% (ms)	75% (ms)
B	-40.00	42.82	-50.03	-29.17
T	-57.09	71.62	61.63	-30.40
V	-107.54	116.02	-130.28	-73.41
E	-252.47	257.47	-279.53	-216.13

Table 5 Results of segmentation without modeling the final segments

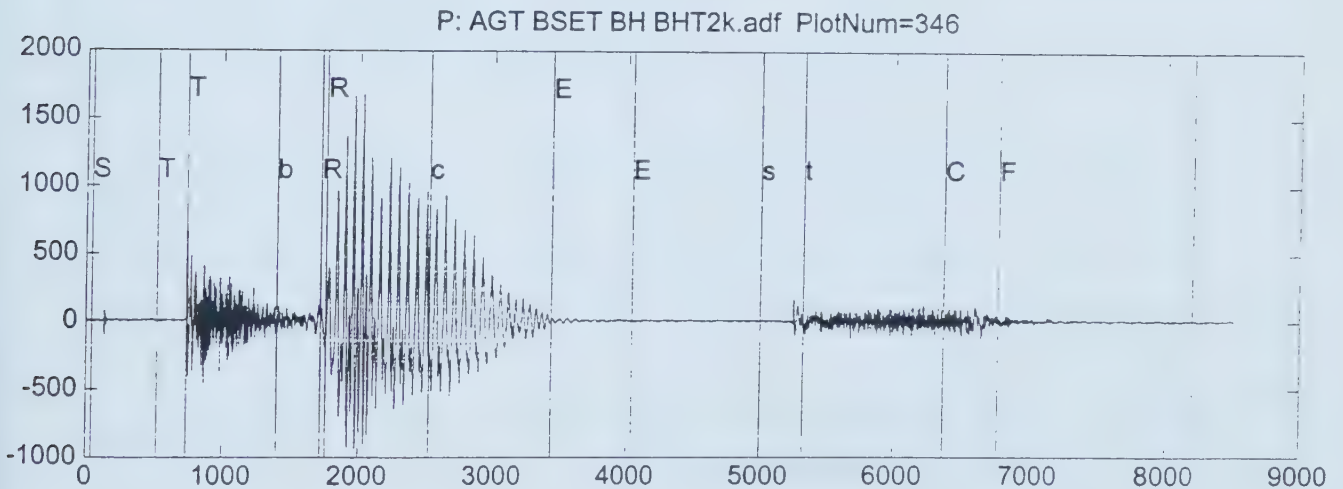


Figure 9 Boundaries for the final consonant

Since there are no hand marked cursors as the reference points, no analysis of correctness is done at this stage. Only a sample plot (Figure 9) is presented to show the resulting boundaries for the final segment.

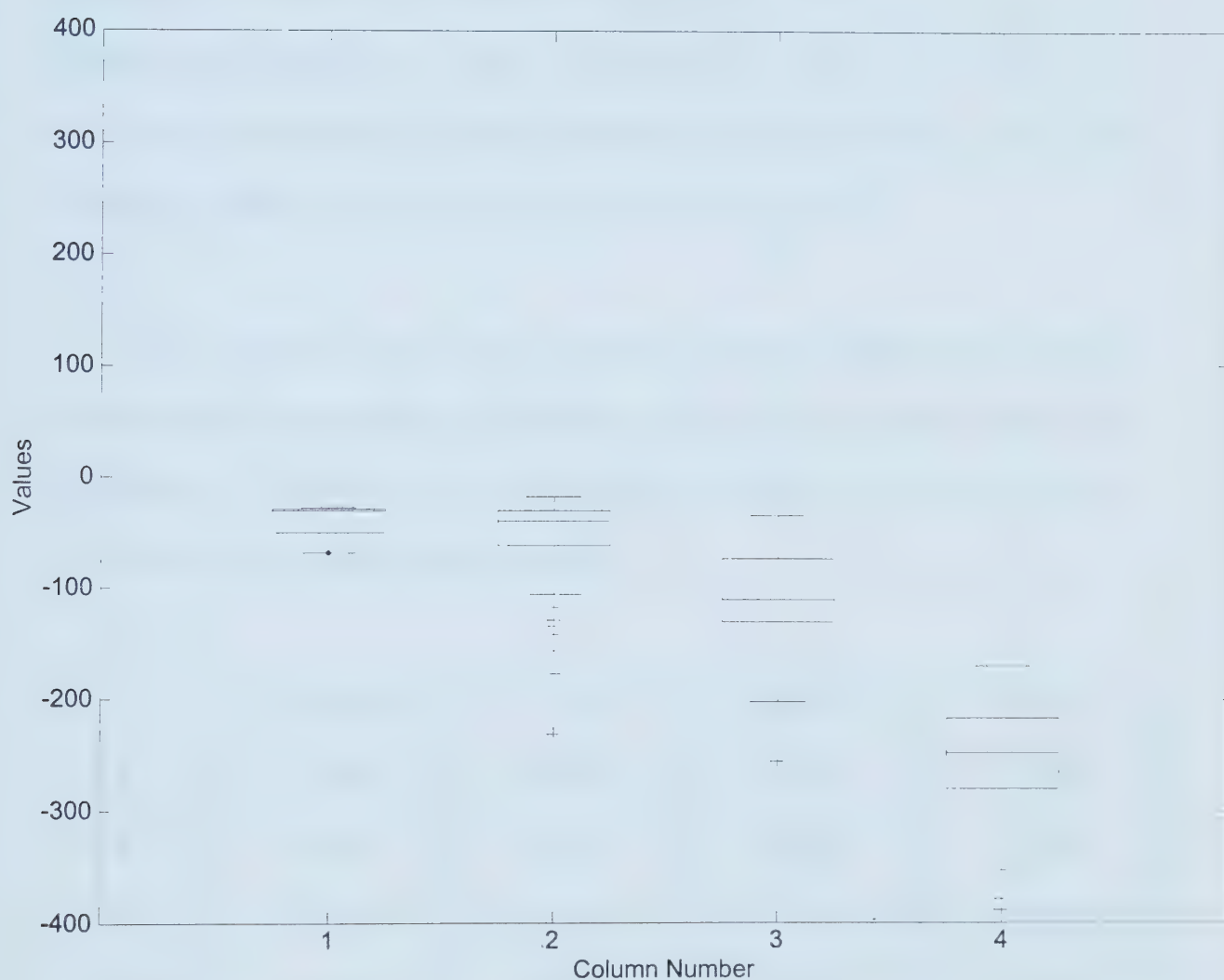


Figure 10 Boxplot of results for Experiment 2 (1-- voice bar, 2 -- burst, 3 -- vowel rise, and 4-- vowel end)

3.3 Experiment 3 -- Reduced Number of Cepstral Coefficients

The previous experiments make use of 13 mel-cepstral coefficients. Experiment 3 varies the number of coefficients to see if 13 is a reasonable number to be adopted in the studies. Experiments are done with several trials of 6, 9, and 13 numbers of cepstral coefficients. Table 6 shows the results with 9 cepstral coefficients and the boxplot is shown in Figure 11.

The results in Table 6 show that the error rate has increased for all microsegments compared to Experiment 1 which uses 13 cepstral coefficients. The increase in rms values for voicebar, burst, vowel rise, and vowel end are: 15.89, 15.20, 61.57, and 94.50 respectively.

	mean(ms)	rms	25% (ms)	75% (ms)
B	-30.98	38.99	-50.03	-6.60
T	-15.70	27.66	-15.22	-7.47
V	19.47	102.36	-9.13	18.56
E	46.16	147.46	2.34	118.59

Table 6 Experiment with 9 cepstral coefficients

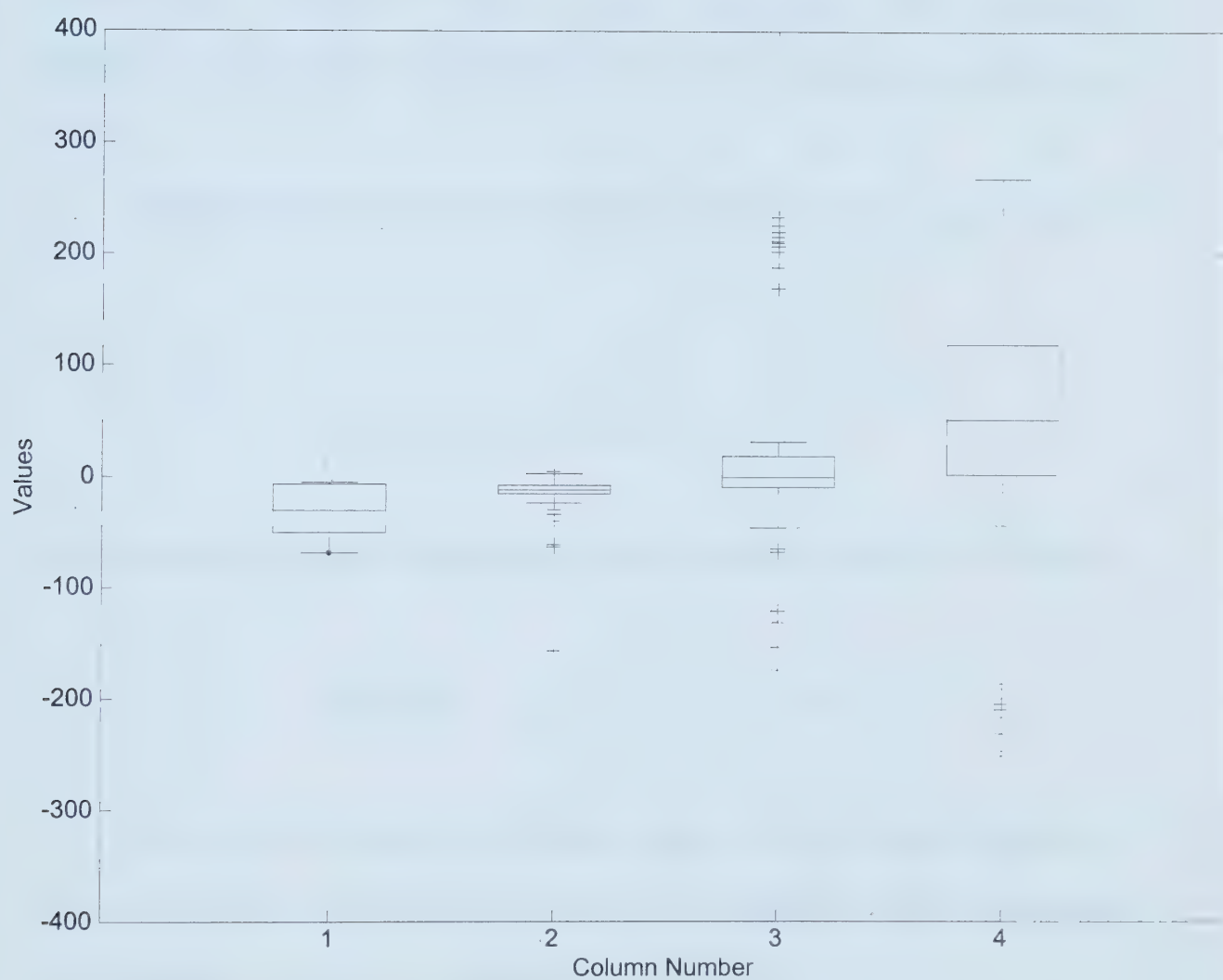


Figure 11 Boxplot of experiment 3 (1-- voice bar, 2 -- burst, 3 -- vowel rise, and 4-- vowel end)

3.4 Experiment 4 -- Introduction of Delta-cepstral Coefficients

Delta-cepstral coefficients can represent the aspects of dynamic change of the signal properties within a single observation. This experiment introduces delta-cepstral coefficients to be used with mel-cepstral coefficients.

Delta cepstral coefficients are calculated using the following formula:

$$d_t = \sum_{\tau=1, \dots, N} \tau(c_{t+\tau} - c_{t-\tau}) / 2 \sum_{\tau=1, \dots, N} \tau^2$$

At the edges of the data, the following simple first order differences are used:

$$d_t = c_{t+1} - c_t, \quad t < N$$

Here c_t is the cepstral coefficient at time t and N is the number of frames over which the delta-coefficients are calculated. In this experiment, seven frames are used for calculating the coefficients.

The model is trained and tested using 13 mel- and 13 delta-cepstral coefficients. The results are in Table 7 and Figure 12 shows the boxplot.

Compared to Experiment 1, the rms value decreases by 3.18 for the voicebar. For the rest of the microsegments, there are increases of 7.91 for burst, 15.18 for vowel rise, and 71.36 for vowel end.

	mean(ms)	rms	25% (ms)	75% (ms)
B	-15.99	19.92	-24.41	-6.50
T	-7.43	20.37	-12.63	3.09
V	-2.27	55.97	-15.23	17.02
E	16.67	124.32	0.97	88.69

Table 7 Experiment with delta-cepstral coefficients

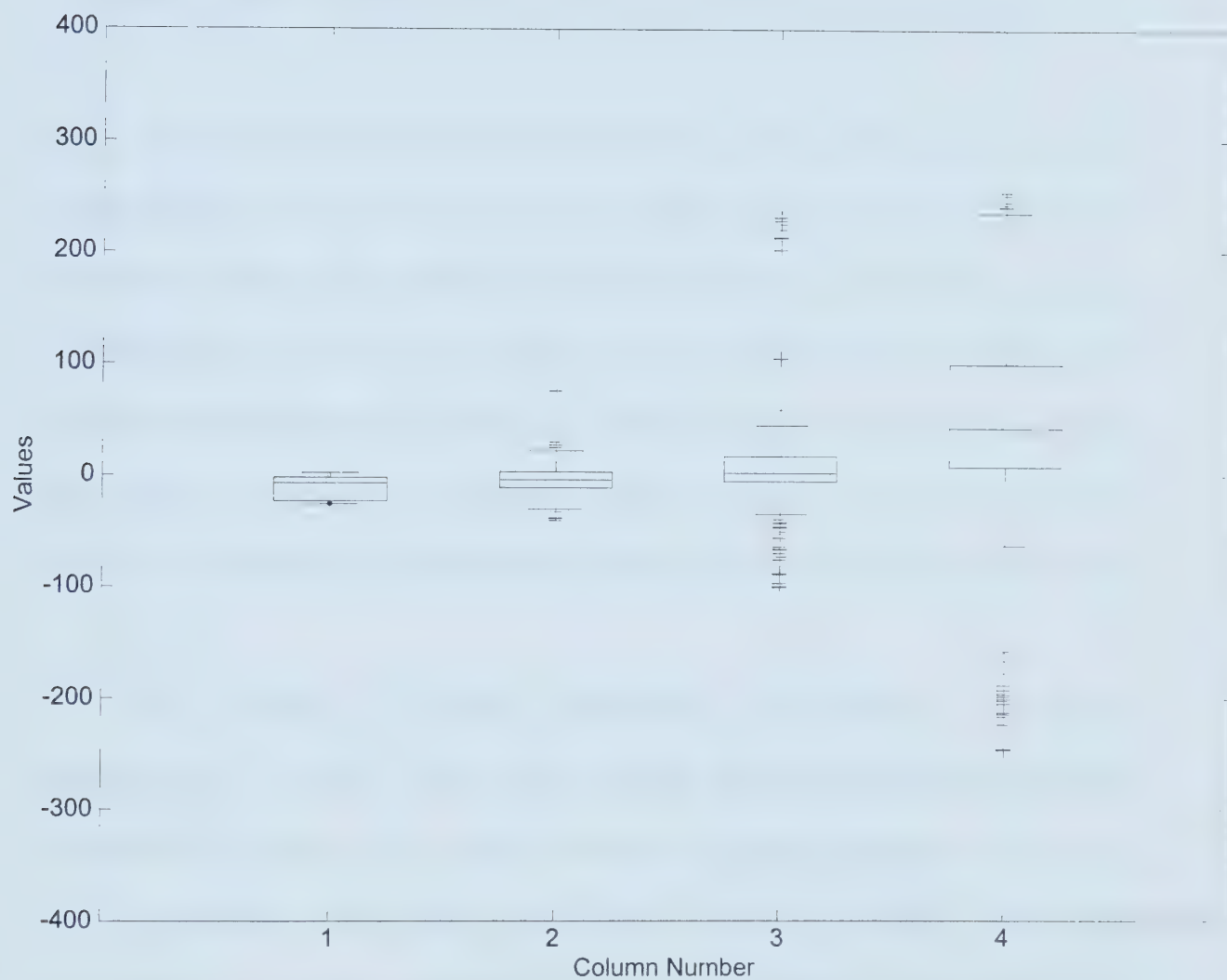


Figure 12 Boxplot of results for experiment 4 (1-- voice bar, 2 -- burst, 3 -- vowel rise, and 4-- vowel end)

4. Discussion and Conclusion

The best results were achieved in Experiment 1. The results are given in Table 4. Figure 8 shows the boxplot of the distribution of the results.

The final /k/ is segmented the same way as the initial consonants. We assume that it consists of the same microsegments which are relatively steady for all stops (Deng, et.al., 1994). The purpose of modeling the final segment is to improve the entire model performance, since the side effects of the final segment modeling may optimize the Viterbi procedure. As we can see from the results of Experiment 2 (Table 5 and Figure 10), the model performance is very poor compared to Experiment 1 which uses the final segment modeling.

The number of cepstral coefficients also influences the model performance. The base value of the number should be set large enough to represent the signal with enough information. On the other hand, if there are more parameters with the same size of training sample, the model performance decreases presumably due to not enough training data. The experiments tested the use of six, nine, and thirteen cepstral coefficients. Table 7 gives the results with nine and Table 4 with thirteen coefficients. Thirteen coefficients appears to be a reasonable choice.

The introduction of delta-cepstral coefficients to be used with the mel-cepstral coefficients did not achieve any real improvement in the experiment. The results may be influenced by several factors. First, introducing additional parameters with the same training data may achieve poor results. A larger set of training data is required in this situation. The continuation parts are relatively stable chunks. Therefore, the dynamic information contained in each microsegment will not play a major part. Finally, the choice of the number of frames for calculating the delta-cepstral coefficients can also affect the results. However, for the onsets, the results are somewhat problematic. Since the onsets involve an abrupt change, modeling the dynamic aspects should achieve results. One possible explanation could be that Viterbi search is a global optimization procedure, therefore other factors may also affect the results.

CHAPTER 4 SUMMARY

This research is the study of an automatic procedure for segmentation of English CVC syllables. It is based on the assumption that phoneme-size segments can be represented by smaller phonetic entities that contain relatively stable chunks which we refer to as microsegments. Therefore, stops, although highly non-stationary in their overall range, consist of some relative stationary fragments. Based on this assumption, we used a modified semi-Hidden Markov model to model the speech sequence, in which we take the advantage of HMM modeling which assumes a sequence of observations to be statistically independent of each other so that elegant mathematical tools can be adopted.

The microsegments are further classified into the onsets and offsets, based on their different acoustic properties, and they are used as states of the model. A simplification procedure is adopted to set the model transition parameters *a priori*. The observations are described by a single mixture multivariate Gaussian function with a full covariance matrix, because it proves that explicitly modeling the correlation of the parameters for the observation achieves better results than modeling it implicitly. The state durations are modified using uniformly distributed function bounded by

minimum and maximum duration constraints. The outputs are obtained through Viterbi search. The viterbi algorithm is a dynamic programming algorithm. It has the property that the remaining decisions must constitute an optimal policy with regard to the state resulting from the decision made before. Therefore, what it does is to temporarily backtrack at each microsegment level and it is 'optimal' in terms of joint maximization of observations. In this sense, each CVC syllable is represented as a sequence of observations.

The task of the segmentation is to mark the boundaries of the microsegments for the CV of the CVC structure. To improve the model performance, the final C is further modeled the same way as the initial segment. The side effects of the final modeling are expected to improve the output results since the Viterbi search is a global optimization procedure.

The implementation of the research takes into consideration the intended applications, technological feasibility, and the theoretical approach. Certain simplifications are made in the modeling procedure for ease of application, which include parameter optimization. Therefore, the resulting system is not a HMM in a strict sense. Rather it follows the track of a dynamic programming process.

The goal of the research is to investigate the effects of variation of certain aspects of the model architecture. Therefore, the segmentation experiments are also conducted under several conditions in order to find out the optimal model architecture. They include adopting different numbers of mel-cepstral coefficients and introducing delta-cepstral coefficients.

Certain further work can be done to increase the model performance. First the model parameters can be optimized in certain ways. If the training sample is large enough, the traditional method of the forward and backward procedure might be adopted to train the model for optimal parameters.

Mixture models may also increase the model performance. A single continuous probability density function associated with each state is usually not enough to model complicated observations. Mixture models are usually required for detailed modeling but this brings about increasing computational complexity.

Also, some normalization measures can be taken for the feature vectors or other representations of feature vectors instead of cepstral coefficients can be used. In Nearey and Kiefte (1995), mean absolute amplitudes are used of the original signal, a high-pass signal, and a band-pass

filtered signal. The maximum mean absolute amplitude over all frames for the entire syllable is also included as a normalization measure.

The advantage of the model adopted in this research is that we make use of acoustic-phonetic knowledge about the segments to obtain composite phonetic entities that represent a sequence of speech events. The size of the model can be greatly reduced compared to the phoneme model because the microsegments are shared across different phonemes or different allophones of the same phoneme. The use of such phonetic units may make large vocabulary systems easier to develop because the microsegments repeat more frequently than phoneme. Therefore, smaller training sample size is possible. Another advantage is the strong discriminability of the model. Since the same model is used for all microsegments of the same type, this eliminates the random variation of a specific microsegment.

REFERENCE

- Burshtein D. (1996). Robust Parametric Modeling of Durations in Hidden Markov Models. *IEEE Transactions on speech and Audio Processing*, 3.4 (May), 240-242.
- Deshayes, J., & Picard, D. (1986). 'Off-line statistical analysis in change-point models using nonparametric and likelihood method,' in Detection of Abrupt Changes in Signals and Dynamical Systems. M. Basseville and A. Benveniste, Eds. New York: Springer-Verlag.
- Furui, S. (1985). "Digital Speech Processing, Synthesis, and Recognition." Tokyo: Tokai University Press.
- Guedon, Y. (1992). Review of several stochastic speech unit models. *Computer Speech and Language*, 6, 377-402.
- Huang, X.D., Ariki, Y. & Jack, M.A. (1990). *Hidden Markov Models for Speech Recognition*. Edinburgh: Edinburgh University Press.
- Ince, N. A. (Ed) (1992). *Digital Speech Processing: Speech Coding, Synthesis and Recognition.*, Norwell: Kluwer Academic Publishers.
- Johnson, R. A. & Wichern, D. W. (1982). *Applied Multivariate Statistical Analysis*. Englewood: Prentice-Hall, Inc.
- Kachigan, S. K. (1991). *Multivariate Statistical Analysis*. New York: Radius Press.
- Kent, R. D. & Read, R. (1992). *The Acoustic Analysis of Speech*. San Diego: Singular Publishing Group, Inc.

- Leung, H. C., & Zue, V. W. (1984). A Procedure for Automatic Alignment of Phonetic Transcriptions with Continuous Speech. *ICASSP*, 1, 2.7.1.-2.7.4.
- Li T.H., & Gibson J. D. (1996). Speech Analysis and Segmentation by Parametric Filtering. *IEEE Transactions on Speech and Audio Processing*, 3.4 (May), 203-213.
- Ljolje, A. (1994). The Importance of Cepstral Parameter Correlations in Speech Recognition. *Computer Speech and Language*, 8, 223-232.
- Ljolje, A., & Levinson, S. E. (1991). Development of an Acoustic-Phonetic Hidden Markov Model for Continuous Speech Recognition. *IEEE Transactions on signal processing*, 39.1 (Jan.), 29-39.
- Ostendorf M., Digalakis V. V., & Kimball O. A. (1996). HMM's to Segment Models: A Unified View of Stochastic Modeling for Speech Recognition. *IEEE Transactions on speech and Audio Processing*, 4.5 (Sept.), 360-378.
- Owens, F.J. (1993). *Signal Processing of Speech..* London: The Macmillan Press Ltd
- Pirani, G. (Ed) ((1990). *Advanced Algorithms and Architectures for Speech Understanding*. Berlin: Springer-Verlag.
- Rabiner, L. & Juang, B. H. (1993). *Fundamentals of Speech Recognition.*, Englewood Cliffs: Prentice Hall PTR.
- Saito, S. (Ed) (1992). *Speech Science and Technology*. Tokyo: Ohmsha, Ltd.
- Svendsen, T., & Soong, F. K. (1987). On the Automatic Segmentation of Speech Signals. *ICASSP*, 1: 77-80.

- Vidal E., & Marzal A. (1990). A Review and New Approaches for Automatic Segmentation of Speech Signals. *Proceedings of EUSIPCO'90*, 43-53.
- Wang, X. (1994). Durationally Constrained Training of HMM without Explicit State Durational PDF. *Proceedings, Institute of Phonetic Sciences, University of Amsterdam*, 18, 111-130.

University of Alberta Library



0 1620 0926 1692

B45645